

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-098659

(43)Date of publication of application : 11.04.1995

(51)Int.Cl.

G06F 9/445

G06F 13/00

G06F 15/177

(21)Application number : 05-243382

(71)Applicant : TEC CORP

(22)Date of filing : 29.09.1993

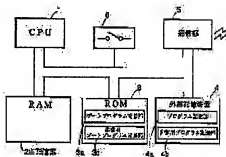
(72)Inventor : OKA TOMOHIDE

(54) DOWN LOAD BOOTSTRAP METHOD

(57)Abstract:

PURPOSE: To easily and speedily change programs in a device executing an unmanned operation and a remote device by automatically down-loading the program and raising it.

CONSTITUTION: CPU 1 executes the program stored in a main storage part 2. At the time of the normal rise of CPU 1, a bootstrap program stored in a boot program storage part 3a is executed, and the program is transferred from a program storage part 4 to the main storage part 2 in a state where CPU 1 can execute it. When abnormality occurs at the rise time of CPU 1, a bootstrap program for emergency stored in an emergency program storage part 3b is executed, and a program for emergency is transferred from an emergency program storage part 4b to the main storage part 2 in the state where CPU 1 can execute it. The program is down-loaded on a storage part 4a from outside through a communication part 5. Then, the bootstrap program is executed.



LEGAL STATUS

[Date of request for examination] 25.03.1997

[Date of sending the examiner's decision of rejection] 17.05.2000

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3296520

[Date of registration] 12.04.2002

[Number of appeal against examiner's decision of rejection] 2000-09077

[Date of requesting appeal against examiner's decision of rejection] 16.06.2000

[Date of extinction of right]

* NOTICES *

JPO and NCIP I are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] By the bootstrap program which memorized the program downloaded through the communications department in a non-volatile and the rewritable program store section, and was memorized in the boot program storage section of a non-volatile It is the download bootstrap approach of transmitting said program to the primary storage memorized in the condition that CPU can be performed. The program store section for emergencies of a non-volatile which memorizes the program for emergencies which downloads a program in said program store section from the exterior through said communications department, The boot program storage section for emergencies which memorizes the bootstrap program for emergencies which transmits said program for emergencies to said primary storage in the condition that said CPU can be performed, from this program store section for emergencies is prepared. Said bootstrap program is performed at the time of a standup with said normal CPU. By this bootstrap program When said program is transmitted to said primary storage from said program store section and abnormalities arise at the time of the standup of said CPU Said bootstrap program for emergencies is performed. By this bootstrap program for emergencies Transmit said program for emergencies to said primary storage from said program store section for emergencies, and by this program for emergencies, while downloading a program in said program store section from the exterior through said communications department The download bootstrap approach characterized by transmitting said program which performed said bootstrap program and was downloaded from said program store section to said primary storage.

[Translation done.]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention relates to the download bootstrap approach which can change a program by download from the outside and can be started.

[0002]

[Description of the Prior Art] In case CPU performs conventionally the program memorized by main storage, such as RAM, a program is memorized by the external storage of the non-volatile of a hard disk drive unit etc., and the bootstrap approach which performs the bootstrap program of the small capacity first memorized to ROM when starting equipment, and transmits a program to RAM from external storage by this bootstrap program is used.

[0003] However, when exchanging ROM which memorized the program by handcraft as mentioned above, while program modification of the equipment which performs uninhabited actuation, and the equipment installed far away is difficult, carrying out for a short time simultaneous to the equipment which had much rewritings of a program installed has the difficult problem.

[0004] Then, when changing a program, the configuration which uses means of communications and downloads a program from the exterior is considered.

[0005]

[Problem(s) to be Solved by the Invention] However, as mentioned above, in downloading a program from the exterior using means of communications, in the unexpected power-source cutoff under download etc., it has the problem to which starting becomes impossible, a manual starting activity is needed, and actuation becomes complicated.

[0006] This invention was made in view of such a point, and aims at offering the download bootstrap approach which can be changed easily and certainly by downloading a program from the outside.

[0007]

[Means for Solving the Problem]

* NOTICES *

JPO and NCIP I are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.*** shows the word which can not be translated.

3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is a flow chart explaining actuation of one example of the equipment using the download bootstrap approach of this invention.

[Drawing 2] It is the block diagram of the equipment using the download bootstrap approach same as the above.

[Description of Notations]

1 CPU

2 RAM Which Constitutes Primary Storage

3a Boot program storage section

3b The boot program storage section for emergencies

4a Program store section

4b The program store section for emergencies

5 Communications Department

[Translation done.]

【特許請求の範囲】

【請求項 1】 通信部を介してダウンロードしたプログラムを不揮発性かつ書き替え可能なプログラム記憶部に記憶し、不揮発性のブートプログラム記憶部に記憶したブートストラッププログラムにより、前記プログラムを CPU が実行可能な状態で記憶する主記憶部に転送するダウンロード・ブートストラップ方法であって、前記通信部を介して外部から前記プログラム記憶部にプログラムをダウンロードする非常用プログラムを記憶する不揮発性の非常用プログラム記憶部と、この非常用プログラム記憶部から前記主記憶部に前記非常用プログラムを前記 CPU が実行可能な状態で転送する非常用ブートストラッププログラムを記憶する非常用ブートプログラム記憶部とを設け、前記 CPU の正常な立ち上がり時には、前記ブートストラッププログラムを実行し、このブートストラッププログラムにより、前記プログラム記憶部から前記主記憶部に前記プログラムを転送し、前記 CPU の立ち上がり時に異常が生じた場合には、前記非常用ブートストラッププログラムを実行し、この非常用ブートストラッププログラムにより、前記非常用プログラム記憶部から前記主記憶部に前記非常用プログラムを転送し、この非常用プログラムにより、前記通信部を介して外部から前記プログラム記憶部にプログラムをダウンロードするとともに、前記ブートストラッププログラムを実行し、前記プログラム記憶部から前記主記憶部にダウンロードした前記プログラムを転送することを特徴としたダウンロード・ブートストラップ方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、外部からのダウンロードによりプログラムを変更して立ち上げることができるダウンロード・ブートストラップ方法に関する。

【0002】

【従来の技術】 従来、CPU は、RAM などの主記憶装置に記憶されたプログラムを実行する際、プログラムは、ハードディスク装置などの不揮発性の外部記憶装置に記憶され、そして、装置を立ち上げる際に、まず ROM に記憶した小容量のブートストラッププログラムを実行し、このブートストラッププログラムにより、外部記憶装置から RAM にプログラムを転送するブートストラップ方法が用いられている。

【0003】 しかしながら、上記のように手作業によりプログラムを記憶した ROM を交換する場合には、無人動作を行う装置や、遠方に設置する装置のプログラム変更が困難であるとともに、プログラムの書き換えを多数設置された装置に対して同時にあるいは短時間に行うことが困難である問題と有している。

【0004】 そこで、プログラムを変更する場合に、通信手段を用いて外部からプログラムをダウンロードする

構成が考えられている。

【0005】

【発明が解決しようとする課題】 しかしながら、上記のように、通信手段を用いて外部からプログラムをダウンロードする場合には、ダウンロード中の不慮の電源遮断などの場合は立ち上げができなくなり、手作業による立ち上げ作業が必要になり操作が煩雑になる問題を有している。

【0006】 本発明は、このような点に鑑みなされたもので、プログラムを外部からダウンロードすることにより、容易かつ確実に変更できるダウンロード・ブートストラップ方法を提供することを目的とする。

【0007】

【課題を解決するための手段】 本発明のダウンロード・ブートストラップ方法は、通信部を介してダウンロードしたプログラムを不揮発性かつ書き替え可能なプログラム記憶部に記憶し、不揮発性のブートプログラム記憶部に記憶したブートストラッププログラムにより、前記プログラムを CPU が実行可能な状態で記憶する主記憶部に転送するダウンロード・ブートストラップ方法であって、前記通信部を介して外部から前記プログラム記憶部にプログラムをダウンロードする非常用プログラムを記憶する不揮発性の非常用プログラム記憶部と、この非常用プログラム記憶部から前記主記憶部に前記非常用プログラムを前記 CPU が実行可能な状態で転送する非常用ブートストラッププログラムを記憶する非常用ブートプログラム記憶部とを設け、前記 CPU の正常な立ち上がり時には、前記ブートストラッププログラムを実行し、このブートストラッププログラムにより、前記プログラム記憶部から前記主記憶部に前記プログラムを転送し、前記 CPU の立ち上がり時に異常が生じた場合には、前記非常用ブートストラッププログラムを実行し、この非常用ブートストラッププログラムにより、前記通信部を介して外部から前記プログラム記憶部にプログラムをダウンロードするとともに、前記ブートストラッププログラムを実行し、前記プログラム記憶部から前記主記憶部にダウンロードした前記プログラムを転送するものである。

【0008】

【作用】 本発明のダウンロード・ブートストラップ方法では、CPU が主記憶部に記憶されたプログラムを実行して装置を動作させる。CPU の正常な立ち上がり時には、ブートプログラム記憶部に記憶されたブートストラッププログラムが実行され、プログラム記憶部から主記憶部に CPU が実行可能な状態でプログラムが転送される。CPU の立ち上がり時に異常が生じた場合には、非常用ブートプログラム記憶部に記憶された非常用ブートストラッププログラムが実行され、非常用プログラム記

憶部から主記憶部に非常用プログラムがCPUが実行可能な状態で転送され、この非常用プログラムにより、通信部を介して外部からプログラム記憶部にプログラムがダウンロードされる。そして、ブートストラッププログラムが実行され、プログラム記憶部から主記憶部にプログラムが転送される。

【0009】

【実施例】以下、本発明のダウンロード・ブートストラップ方法の一実施例の構成を図面を参照して説明する。

【0010】図2に示すように、CPU1に、主記憶部10を構成するRAM2、ROM3、外部記憶装置4、通信部5、および設定スイッチ6などが接続され、例えば、発券機を制御する制御装置が構成されている。

【0011】そして、CPU1は、RAM2上に記憶されたデータを用い、同じくRAM2上に記憶されたプログラムを実行して、装置を動作させる。

【0012】また、ROM3上には、ブートプログラム記憶部3aと、非常用ブートプログラム記憶部3bとが割り当てられ、ブートプログラム記憶部3aには、CPU1の通常の立ち上がり時に実行されるブートストラッププログラムが記憶され、非常用ブートプログラム記憶部3bには、異常時に実行される非常用ブートストラッププログラムが記憶されている。そして、電源投入時やリセット動作時などに、CPU1がこのブートストラッププログラムを実行し、外部記憶装置4から読み込んだプログラムをRAM2に書き込むようになっている。

【0013】さらに、外部記憶装置4は、ハードディスク、フレキシブルディスク、光磁気ディスク、電気的にメモリ内容を書き込み・消去できるEEPROM、不揮発性RAM (nonvolatile RAM)、バッテリーによりバックアップされたRAMなど、書き替え可能で、かつ、電源を切っても内容が消失しない不揮発性の記憶装置により構成されている。そして、CPU1は、この通信部5に、プログラム記憶部4aおよび非常用プログラム記憶部4bが割り当てられ、それぞれ、ハフマン法、LZ法などのデータ圧縮方法により圧縮された状態で、あるいは非圧縮状態で、発券機を制御するプログラムと、立ち上げに異常が生じた際に用いるための非常用プログラムとが記憶されている。

【0014】また、通信部5は、公衆電話回線あるいは専用回線などを通じて制御装置外部のホストコンピュータに接続されている。そして、CPU1は、この通信部5を介して、ホストコンピュータから送信されたプログラムおよびデータなどを外部記憶装置4にダウンロードするようになっている。

【0015】さらに、設定スイッチ6は、オン・オフを切り替えることにより、正常側と異常側とを切り替えることが可能で、この設定スイッチ6を正常側に設定した状態で、CPU1の立ち上がり時に、通常のブートストラッププログラムが実行されるとともに、この設定ス

イッチ6を異常側に設定した状態で、強制的に非常用ブートストラッププログラムが実行されるようになっていく。

【0016】次に、図1のフローチャートを参照して、本実施例の動作を説明する。

【0017】まず、設定スイッチ6を正常側または異常側のいずれかに設定した状態で、電源を投入する(ステップ1)。

【0018】そして、設定スイッチ6が正常側に設定されている場合には(ステップ2)、まずCPU1がROM3に記憶された通常用のブートストラッププログラムを実行し、このブートストラッププログラムに従って、外部記憶装置4のプログラム記憶部4aに記憶されたプログラムおよびデータを読み込み、CPU1が直接実行可能なRAM2上に転送する(ステップ3)。

【0019】このとき、プログラムおよびデータが圧縮状態で記憶されているかを判断し(ステップ4)、圧縮されている場合にはこれらのプログラムを伸長(解凍)し(ステップ5)、圧縮されていない場合にはそのまま、RAM2上に書き込む。

【0020】また、チェックサム法、パリティ法、またはLRC法などによるデータ誤りの検出を行い(ステップ6)、誤りが検出されず正常な場合には(ステップ7)、ブートストラッププログラムを終了する。この状態で、RAM2上に展開されたプログラムの実行が開始される(ステップ8)。

【0021】そして、このプログラムを実行する過程において、プログラムの書き替えを行わない場合には(ステップ9)、通常の発券機の動作を開始する(ステップ10)。

【0022】また、この発券機を制御するプログラムを実行する過程において、プログラムにより自動的に、または、設定スイッチ6あるいは図示しないキーボードを操作することにより、あるいは、通信部5を介した外部からの命令により、プログラムの書き替えを行なう場合には(ステップ9)、通信部5が制御され、外部との通信が開始される(ステップ11)。そして、この通信部5を介して、外部から外部記憶装置4のプログラム記憶部4aに圧縮状態あるいは非圧縮状態のプログラムがダウンロードされ、外部記憶装置4のプログラムが変更、更新される(ステップ12)。

【0023】そして、プログラムの変更が終了した状態で(ステップ13)、ブートストラッププログラムが再び実行され、このブートストラッププログラムに従って、外部記憶装置4aに記憶された新しいプログラムおよびデータをRAM2上に転送する(ステップ3~7)。

【0024】続いて、RAM2上に展開されたプログラムの実行が開始され、通常の発券機の動作を開始する(ステップ8~10)。

【0025】一方、設定スイッチ6が異常側に設定され

ている場合(ステップ2)、およびデータ誤りの検出過程において誤りが検出されて、プログラムまたはデータの展開が正常に終了しなかった場合には(ステップ6, 7)、CPU1がROM3の非常用ブートプログラム記憶部3bに記憶された非常用ブートスト랩プログラムを実行し、この非常用ブートスト랩プログラムに従って、外部記憶装置4の非常用プログラム記憶部4bに記憶された非常用プログラムを読み込み、圧縮データであればこれを伸長しながらRAM2上に転送する(ステップ14~16)。

【0026】ついで、このRAM2に転送された非常用プログラムが実行され(ステップ17)、通信部5が制御されて、外部との通信が開始される(ステップ18)。そして、この通信部5を介して、外部から外部記憶装置4のプログラム記憶部4aに圧縮状態あるいは非圧縮状態のプログラムがダウンロードされる(ステップ19)。

【0027】そして、外部記憶装置4へのプログラムの書き込みが終了した状態で(ステップ20)、ブートスト랩プログラムが再び実行され、このブートスト랩プログラムに従って、外部記憶装置4に記憶された新しいプログラムおよびデータを読み込み、RAM2上に展開する(ステップ3~7)。

【0028】続いて、RAM2上に展開されたプログラムの実行が開始され、通常の発動動作を開始する(ステップ8~10)。

【0029】そして、本実施例によれば、手作業による設定スイッチ6の操作のほか、自動的にプログラムの変更を行うことが可能なため、無人動作を行う装置や、遠方に設置する装置についても、容易かつ迅速にプログラムの変更を行うことができる。

【0030】また、プログラムの変更は、外部からのダウンロードにより行われるため、多数設置された装置に対しても、プログラムの書き換えを同時にあるいは短時間に行うことができる。

【0031】さらに、ダウンロード中に、バッテリーまたはバックアップ電源などの出力低下による不慮の電源遮断や通信遮断が生じ、あるいは、装置不良や電源線からのノイズなどによるデータの変化(文字化け)や欠落(文字落ち)などが生じた場合には、自動的に誤りが検出され、非常用ブートスト랩プログラムが実行されてRAM2に非常用プログラムが読み込まれ、この非常用プログラムによりプログラムを外部からダウンロード

することができる。そこで、手作業による立ち上げ作業の必要がなく、確実にプログラムを変更することができる。

【0032】また、誤りが検出されなくとも、プログラムあるいはデータに不良が生じた場合や、強制的にプログラムの更新を行う場合には、設定スイッチ6を操作す

ることにより、強制的に非常用ブートスト랩プログラムおよび非常用プログラムを実行し、プログラムの変更を行うことができる。

【0033】さらに、プログラムは外部からのダウンロードにより変更できるとともに、外部記憶装置4には、圧縮状態のプログラムを記憶し、RAM2への転送時にデータの伸長および誤り検出を同時に行うことにより、外部記憶装置4には、最低限の容量のプログラム記憶部4aを割り当てるとともに、高価な外部記憶装置を小型化できるとともに安価に構成することができる。

【0034】なお、上記の実施例では、非常用プログラム記憶部4bを外部記憶装置4に割り当てたが、この非常用プログラム記憶部4bをROM3上に割り当て、あるいは、ROM3および外部記憶装置4とは別個の不揮発性の記憶装置に記憶させることもできる。

【0035】

【発明の効果】本発明のダウンロード・ブートスト랩方法によれば、自動的にプログラムがダウンロードされて立ち上げられるため、無人動作を行う装置や、遠方に設置する装置についても、容易かつ迅速にプログラムの変更を行うことができるとともに、多数設置された装置に対しても、プログラムの書き換えを同時にあるいは短時間に行うことができる。さらに、ダウンロードの異常などにより、ブートスト랩プログラムの実行中に異常が生じた場合には、非常用ブートスト랩プログラムが実行されて主記憶部に非常用プログラムが読み込まれ、この非常用プログラムによりプログラムを外部からダウンロードできるため、手作業による立ち上げ作業の必要がなく、確実にプログラムを変更することができるとともに、プログラムは外部からのダウンロードにより変更できるため、最低限の容量のプログラム記憶部を設定するのみで良く、記憶装置の記憶容量を削減して、装置を安価に構成することができる。

【図面の簡単な説明】

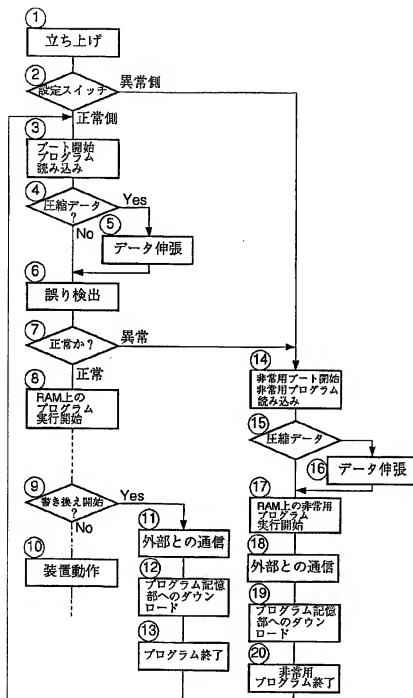
【図1】本発明のダウンロード・ブートスト랩方法を用いた装置の一実施例の動作を説明するフローチャートである。

【図2】同上ダウンロード・ブートスト랩方法を用いた装置のブロック図である。

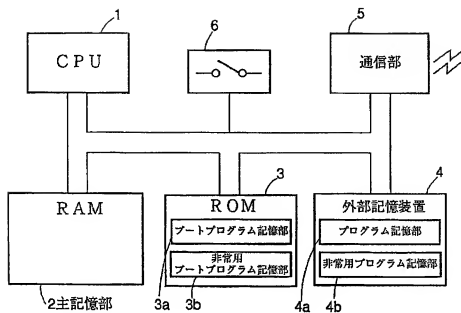
【符号の説明】

- 1 CPU
- 2 主記憶部を構成するRAM
- 3a ブートプログラム記憶部
- 3b 非常用ブートプログラム記憶部
- 4a プログラム記憶部
- 4b 非常用プログラム記憶部
- 5 通信部

【図 1】



【図 2】



PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-027953

(43)Date of publication of application : 30.01.2001

(51)Int.Cl.

G06F 9/445

(21)Application number : 11-201617

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 15.07.1999

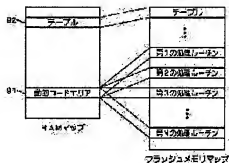
(72)Inventor : YOSHIMURA YOSHIMASA

(54) SEMICONDUCTOR STORAGE DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide an external storage device loading firmware exceeding the size of RAM of a prescribed size while the RAM of a prescribed size is adopted as a memory storing firmware.

SOLUTION: Since a flash memory group can be accessed in a block unit, plural program codes corresponding to different processing routines can individually be accessed. RAM has a first storage area 91 being a dynamic loading area and the program codes stored for each of the blocks in the flash memory group are exclusively loaded in a first storage area 91. Since the plural program codes corresponding to the different processing routines are exclusively loaded in the first storage area 91, a size requested by RAM is required to be small. Since the flash memory group can be rewritten in firmware, correction and version-up can be realized. In such a case, the size requested by RAM is not increased.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

* NOTICES *

JPO and NCIPi are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.*** shows the word which can not be translated.

3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The non-volatile semi-conductor storage section which two or more program codes corresponding to respectively different processing are stored according to an individual, can rewrite, and has two or more accessible blocks according to an individual, RAM which has the 1st storing field where said two or more program codes are stored exclusively mutually, A semiconductor memory equipped with the error correction processing section transmitted to Above RAM, intervening between said RAM and said non-volatile semi-conductor storage section, and performing error correction processing to one of said two or more of the program codes.

[Claim 2] Said RAM is a semiconductor memory according to claim 1 which has further the 2nd storing field where each start address of a block of said plurality is stored.

[Claim 3] Each of two or more of said program codes is a semiconductor memory according to claim 1 or 2 corresponding to the command given from the outside of said semiconductor memory.

[Translation done.]

* NOTICES *

JPO and NCIPJ are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the RAM disk equipment using a flash memory, and a flash plate ATA card, concerning a semiconductor memory.

[0002]

[Description of the Prior Art] As external storage of a computer, RAM disk equipment, a flash plate ATA (AT attachment) card, etc. are proposed from the former. These are equipped with the controller for controlling the mass flash memory for data accumulation, and them.

[0003] The controller is equipped with the memory for firmware storing for storing the program which are CPU (central processing unit) and a firmware. As memory for firmware storing, if ROM (read-only memory) is adopted, whenever it raises correction and version of a firmware, it is necessary to rebrill ROM, and there is a fault which cannot correct a firmware easily. Therefore, the firmware is beforehand stored in a part of flash memory, and these are loaded to RAM (random access memory) at a power up, and the technique of adopting RAM as memory for firmware storing of performing the program loaded to RAM is proposed.

[0004] Drawing 4 is the conceptual diagram having shown the relation between the RAM map at the time of adopting the above-mentioned approach, and a flash plate memory map. The firmware stored in the flash memory as was shown in the flash plate memory map is loaded to RAM, as shown in a RAM map at a power up. By such approach, there is a merit that correction and version up of a firmware can be easily performed by rewriting the firmware stored in a flash memory.

[0005]

[Problem(s) to be Solved by the Invention] However, when adopting the above-mentioned approach, sufficient size which stores a firmware as a capacity of RAM must be prepared beforehand. A program size expands by correction, a functional addition, etc. of a firmware, and when it cannot finish going into RAM of the capacity currently prepared beforehand, that corrected firmware cannot be carried in this external storage. However, since the size of RAM influences cost, it has a demand of wanting to make it as small as possible.

[0006] Although this invention was made under the above-mentioned background and adopts RAM of fixed size as memory for firmware storing, it aims at *(ing) offering the external storage which can carry the firmware exceeding the size, with realizing a system also with easy low-pricing and correction of a highly efficient firmware.

[0007] In addition, the technique which loads two or more program codes selectively is indicated in JP,1-121933,A.

[0008]

[Means for Solving the Problem]

* NOTICES *

JPO and NCIPJ are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram showing the configuration of the semiconductor memory concerning the gestalt of operation of this invention.

[Drawing 2] It is the conceptual diagram having shown the relation of the RAM map and flash plate memory map in the gestalt of operation of this invention.

[Drawing 3] It is the flow chart which shows actuation of the gestalt of operation of this invention.

[Drawing 4] It is the conceptual diagram having shown the relation of the RAM map and flash plate memory map in a Prior art.

[Description of Notations]

2 RAM, 4 A flash memory group, 41-4m A flash memory, the 5 buffer RAM, 6 An error correction processing circuit, 91 The 1st storing field, 92 2nd storing field.

[Translation done.]

| | | | |
|----------------------------|------|-----------|-------------|
| (51) Int. Cl. ⁷ | 識別記号 | F I | テマコード (参考) |
| G06F 9/445 | | G06F 9/06 | 420 T 5B076 |
| | | | 420 L |

審査請求 未請求 請求項の数 3 O L (全 5 頁)

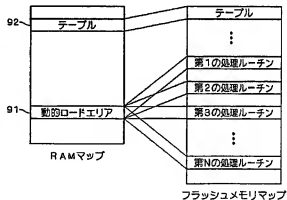
| | | | |
|-----------|--------------------------|----------|---|
| (21) 出願番号 | 特願平11-201617 | (71) 出願人 | 000006013 三菱電機株式会社 東京都千代田区丸の内二丁目2番3号 |
| (22) 出願日 | 平成11年7月15日 (1999. 7. 15) | (72) 発明者 | 吉村 芳正 東京都千代田区丸の内二丁目2番3号 三 菱電機株式会社内 |
| | | (74) 代理人 | 100089233 弁理士 吉田 茂明 (外2名) Fターム (参考) 5B076 BA04 BB12 BB17 CA02 |

(54) 【発明の名称】 半導体記憶装置

(57) 【要約】

【課題】 一定のサイズのRAMをファームウェア格納用メモリとして採用しつつも、そのサイズを超えるファームウェアを搭載することができる外部記憶装置を提供する。

【解決手段】 フラッシュメモリ群はブロック単位でアクセス可能なので、異なる処理ルーチンに対応した複数のプログラムコードは個別にアクセス可能となる。そしてRAMは動的ロードエリアである第1の格納領域91を有しており、第1の格納領域91にはフラッシュメモリ群にブロック毎に格納されていたプログラムコードが排他的にロードされる。従って異なる処理ルーチンに対応した複数のプログラムコードを第1の格納領域91へと排他的にロードするので、RAMに要求されるサイズが小さくて済む。これらのファームウェアは、フラッシュメモリ群が書き換え可能であるので、その修正やバージョンアップが可能である。この場合にRAMに要求されるサイズが増大することもない。



91 : 第1の格納領域

92 : 第2の格納領域

【特許請求の範囲】

【請求項1】 各々異なる処理に対応した複数のプログラムコードが個別に格納され、書き換え可能で個別にアクセス可能な複数のブロックを有する不揮発性半導体記憶部と、

前記複数のプログラムコードが互いに排他的に格納される第1の格納領域を有するRAMと、

前記RAMと前記不揮発性半導体記憶部の間に介在し、前記複数のプログラムコードの一つに対して誤り訂正処理を行ういつつ前記RAMへと転送する誤り訂正処理部とを備える半導体記憶装置。

【請求項2】 前記RAMは、前記複数のブロックの各々の先頭アドレスが格納される第2の格納領域を更に有する、請求項1記載の半導体記憶装置。

【請求項3】 前記複数のプログラムコードの各々は、前記半導体記憶装置の外部から与えられるコマンドに対応する、請求項1又は2記載の半導体記憶装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 この発明は、半導体記憶装置に関し、例えばフラッシュメモリを用いた半導体ディスク装置、フラッシュATAカードに関する。

【0002】

【従来の技術】 従来からコンピュータの外部記憶装置として、半導体ディスク装置やフラッシュATA(AT attachment)カード等が提案されている。これらは、データ蓄積用の大容量フラッシュメモリとそれらを制御するためのコントローラとを備えている。

【0003】 コントローラはCPU (central processing unit)、ファームウェアであるプログラムを格納するためのファームウェア格納用メモリを備えている。ファームウェア格納用メモリとして、ROM (read-only memory)を採用したならば、ファームウェアの修正やバージョンをアップする度にROMを焼き直す必要がある。容易にファームウェアの修正を行えない欠点がある。そのため、フラッシュメモリの一部に予めファームウェアを格納しておき、電源投入時にこれらをRAM(random access memory)にロードし、そしてRAMにロードされたプログラムを実行するという、ファームウェア格納用メモリとしてRAMを採用する手法が提案されている。

【0004】 図4は上記の方法を採用した場合のRAMマップとフラッシュメモリマップとの関係を示した概念図である。フラッシュメモリマップに示されるようにしてフラッシュメモリに格納されていたファームウェアは、電源投入時にRAMマップに示されるようにRAMへロードされる。このような方法ではフラッシュメモリに格納するファームウェアを書き換えることで、容易にファームウェアの修正やバージョンアップを行えるというメリットがある。

【0005】

【発明が解決しようとする課題】 しかし、上記の方法を採用する場合には、RAMの容量としてはファームウェアを格納するだけの十分なサイズを予め用意しておくなければならない。もしファームウェアの修正や機能追加などによってプログラムサイズが膨張し、予め用意されていた容量のRAMに入りきれない場合は、その修正したファームウェアをこの外部記憶装置に搭載することができない。しかしRAMのサイズはコストに影響するので可能な限り小さくしたいという要求がある。

【0006】 本発明は上記の背景のもとでなされたもので、一定のサイズのRAMをファームウェア格納用メモリとして採用しつつも、そのサイズを超えるファームウェアを搭載することができる外部記憶装置を提供し、以て低価格化、高性能のファームウェアの修正も容易なシステムを実現することに資することを目的としている。

【0007】 なお複数のプログラムコードを選択的にロードする技術が例えば特開平1-121933号公報において開示されている。

【0008】

【課題を解決するための手段】 この発明のうち請求項1にかかものものは、各々異なる処理に対応した複数のプログラムコードが個別に格納され、書き換え可能で個別にアクセス可能な複数のブロックを有する不揮発性半導体記憶部と、前記複数のプログラムコードが互いに排他的に格納される第1の格納領域を有するRAMと、前記RAMと前記不揮発性半導体記憶部の間に介在し、前記複数のプログラムコードの一つに対して誤り訂正処理を行ういつつ前記RAMへと転送する誤り訂正処理部とを備える半導体記憶装置である。

【0009】 この発明のうち請求項2にかかものものは、請求項1記載の半導体記憶装置であって、前記RAMは、前記複数のブロックの各々の先頭アドレスが格納される第2の格納領域を更に有する。

【0010】 この発明のうち請求項3にかかものものは、請求項1又は2記載の半導体記憶装置であって、前記複数のプログラムコードの各々は、前記半導体記憶装置の外部から与えられるコマンドに対応する。

【0011】

【発明の実施の形態】 図1は、本発明の一実施の形態にかかる半導体記憶装置として例示される、フラッシュATAカード100の構成を示すブロック図である。フラッシュATAカード100は、ATAバス51を介して外部と接続されており、ATAバス51に接続されるホストインターフェース7を備えている。ATAバス51及びホストインターフェース7を介し、外部からホストコンピュータによってコマンドがATAバス51に入力され、あるいはATAバス51から外部のホストコンピュータへのデータの出力が行われる。

【0012】 フラッシュATAカード100は更に、ホ

特開平7-98659

(43) 公開日 平成7年(1995)4月11日

| | | |
|----------------------------|---------------|------------------------------|
| (51) Int. Cl. ⁶ | 識別記号 | F I |
| G06F 9/445 | | |
| 13/00 | 351 H 7368-5B | |
| 15/177 | | |
| | 9367-5B | G06F 9/06 420 T |
| | 7429-5L | 15/16 420 S |
| | | 審査請求 未請求 請求項の数 1 O L (全 6 頁) |

(21) 出願番号 特願平5-243382

(22) 出願日 平成5年(1993)9月29日

(71) 出願人 00003562

株式会社テック

静岡県田方郡大仁町大仁570番地

(72) 発明者 岡 知英

静岡県三島市南町6番78号 東京電気株式

会社三島工場内

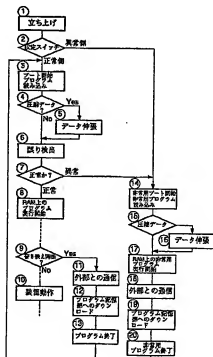
(74) 代理人 弁理士 柳澤 襄 (外2名)

(54) 【発明の名称】ダウンロード・ブートストラップ方法

(57) 【要約】

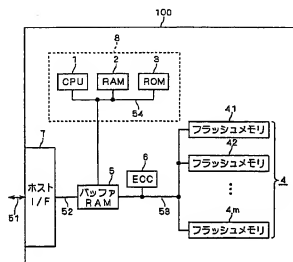
【構成】 CPUがRAMに記憶したプログラムを実行して装置を制御する際、CPUの正常な立ち上がり時には、ブートプログラム記憶部に記憶したブートストラッププログラムを実行し、プログラム記憶部からRAMにCPUが実行可能な状態でプログラムを展開する。CPUの立ち上がり時に異常が生じた場合および設定スイッチを設定した場合には、非常用ブートプログラム記憶部に記憶した非常用ブートストラッププログラムを実行し、非常用プログラム記憶部からRAMに非常用プログラムをCPUが実行可能な状態で展開し、この非常用プログラムにより、通信部を介して外部からプログラム記憶部にプログラムをダウンロードする。続いて、ブートストラッププログラムを実行し、プログラム記憶部からRAMにプログラムを展開する。

【効果】 容易かつ迅速にプログラムを変更できる。記憶装置の容量を削減できる。

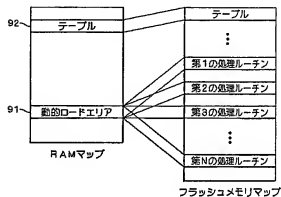


2 RAM、4 フラッシュメモリ群、41~4m フラッシュメモリ、5 バッファRAM、6 誤り訂正処理回路、91 第1の格納領域、92 第2の格納領域。

【図1】



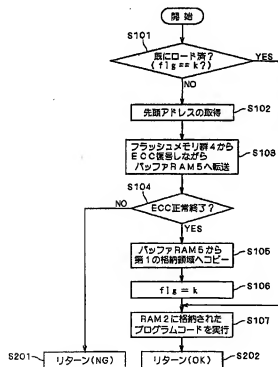
【図2】



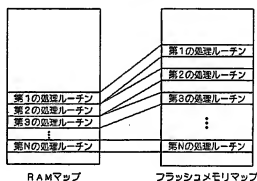
91 : 第1の格納領域

92 : 第2の格納領域

【図3】



【図4】



ストバス52を介してホストインターフェース7に接続されるバッファRAM5と、フラッシュバス53を介してバッファRAM5に接続されるフラッシュメモリ群4とを備えている。また、CPUバス54を介してバッファRAM5に接続されるコントローラ8も備えられている。

【0013】フラッシュメモリ群4は大容量のデータを蓄積する用途に達するため、ブロック（セクタ）単位でアクセス可能な構成を有している。例えば複数のフラッシュメモリ41、42、…、4mから構成されている。もちろんブロック毎にアクセス可能であれば、単数のフラッシュメモリで構成してもよい。コントローラ8はCPUバス54によって互いに接続されたCPU1、RAM2、ROM3を備えている。

【0014】図2はフラッシュATAカード100におけるRAMマップとフラッシュメモリマップとの関係を示した概念図である。フラッシュメモリマップに示されるようにフラッシュメモリに格納されたファームウェアは、各々異なる第1～第Nの処理ルーチンに対応した複数のプログラムコードで構成されており、ブロック毎に格納されている。便宜上、図においてはプログラムコードが格納されている箇所を、その対応する処理ルーチンで示している。

【0015】フラッシュメモリ群4はブロック単位でアクセス可能なため、異なる処理ルーチンに対応した複数のプログラムコードは個別にアクセス可能となる。そしてRAM2は動的ロードエリアである第1の格納領域91を有しており、第1の格納領域91にはフラッシュメモリ群4にブロック毎に格納されていたプログラムコードが排他的にロードされる。このようなフラッシュメモリ群4からRAM2へのプログラムコードの転送は、ROM3に予め格納されたプログラムに基づいてCPU1によって実行させることができる。

【0016】従って本発明によれば、異なる処理ルーチンに対応した複数のプログラムコードを第1の格納領域91へと排他的にロードするので、図4に示される場合と比較してRAM2に要求されるサイズが小さくて済む。そしてもちろん、これらのファームウェアは、フラッシュメモリ群4が書き換え可能であるので、その修正やバージョンアップが可能である。この場合にRAM2に要求されるサイズが増大することもない。新たに追加されるべき処理ルーチンに対応するプログラムコードが、それまでフラッシュメモリ群4においてブロック毎に格納されていたプログラムコードよりも大きい場合であっても、追加されるべき処理ルーチンを分割してブロック毎にフラッシュメモリ群4に格納することにより、RAM2のサイズを変更する必要はない。

【0017】またRAM2は第2の格納領域92を有している。この場合にはフラッシュメモリ群4には第1～第Nの処理ルーチンに対応したプログラムコード

を格納するブロックの、先頭アドレスもテーブルとして格納されている。そしてこのテーブルがRAM2の第2の格納領域92上に格納される。このテーブルの格納は、例えば電源オン時に一度行えば足り、異なるプログラムコードをロードする度にその先頭アドレスをフラッシュメモリ群4から読み出す時間が省略され、迅速な動作を得ることができる。このようなフラッシュメモリ群4からRAM2へのテーブルの転送も、ROM3に予め格納されたプログラムに基づいてCPU1によって実行させることができる。

【0018】なおフラッシュメモリ群4からRAM2へとプログラムコードを転送する際に誤り訂正処理を行うことが望ましい。例えばフラッシュバス53に誤り訂正処理（ECC：error checking and correcting）回路6を接続しておき、フラッシュメモリ群4から呼び出されたプログラムコードに誤り訂正処理を行いながらバッファRAM5に一旦格納する。そしてプログラムコードに訂正不可能な誤りがない場合にのみ、バッファRAM5からRAM2へと当該プログラムコードが転送される。これによって訂正不可能な誤りが発生したプログラムコードをRAM2へロードすることを排除できる。このような動作も、ROM3に予め格納されたプログラムに基づいてCPU1によって実行させることができる。

【0019】バッファRAM5に必要な容量は最もサイズの大きいプログラムコード一つ分足りる。またバッファRAM5はCPU1や外部のホストコンピュータからアクセスで、訂正不可能な誤りが発生したプログラムコードをCPU1や外部のホストコンピュータによって解析することも可能である。

【0020】もちろん、ROM3やRAM2が上記の説明以外のプログラムコードやデータを格納していてもよい。

【0021】以下、詳細な動作について説明する。図3はフラッシュATAカード100の動作のうち、ファームウェアをRAM2にロードする動作を示すフローチャートである。ここではファームウェアのうち、第kの処理ルーチンに対応するプログラムコードをロードする場合を示す。ここで処理ルーチンの番号kは1乃至Nの値を探りうる。

【0022】まずステップS101において、動的ロードエリアたる第1の格納領域91に、第kの処理ルーチンに対応するプログラムコードが格納されているか否かを確認する。具体的には既にロードされているプログラムコードの対応する処理ルーチンの番号が、これから実行しようとする処理ルーチンの番号kと一致しているか否かをチェックする。

【0023】格納されていれば新たなロードは必要ないので、ステップS107へと進み、第2の格納領域92に格納されているプログラムコードをCPU1が実行する。具体的にはアドレスを第1の格納領域91にジャン

ブさせてCPU1がプログラムコードを実行する。その後ステップS202へと進み、他の処理に移行するためにリターンする。この際リターン先のメインルーチンにおいてプログラムコードが正常に格納されたことを示す情報も伝達される(図中のステップS202では“OK”で示している)。

【0024】ステップS101において、第1の格納領域91に第kの処理ルーチンに対応するプログラムコードが格納されていないと判断されれば、新たにプログラムコードを格納する必要がある。そこでステップS102へと進み、第2の格納領域92に格納されたテーブルを参照する。そしてこのテーブルから第kの処理ルーチンに対応するプログラムコードが格納されているブロックの先頭アドレスを取得する。

【0025】次にステップS103に進み、取得された先頭アドレスから始まるブロックから、バッファRAM5にECC処理(誤り訂正処理)を行いながら転送する。そしてステップS104において、誤り訂正処理が正常に終了したかどうか判断される。正常に終了しなかった場合にはステップS201へと進み、他の処理に移行するためにリターンする。この際リターン先のメインルーチンにおいてプログラムコードが正常に格納されなかったことを示す情報も伝達される(図中のステップS201では“NG”で示している)。

【0026】正常に終了した場合にはステップS105へと進み、バッファRAM5に格納されていたプログラムコードを第1の格納領域91へコピーする。そしてステップS106においてフラグfgに処理ルーチンの番号kを代入する(図では代入する“=”で示している)。更にステップS107へと進み、第2の格納領域92に格納されているプログラムコードをCPU1が実行する。その後ステップS202へと進み、他の処理に移行するためにリターンする。

【0027】なおステップS101において、ステップS106で値が決定されたフラグfgと、これから実行しようとする処理ルーチンの番号kと比較することにより(図では比較を“=”で示している)、既にロードされているプログラムコードの対応する処理ルーチンが、所望のものであるかを判断することができる。

【0028】またステップS201、202のリターン先は、図3で示されるフローチャートをするルーチン先を呼び出したアドレスポインタである。

【0029】フラッシュメモリ群4に個別に格納されるプログラムコードが対応する処理ルーチンは、それぞれ例えばATAコマンドの番号に対応させてもよい。つまりホストコンピュータからのATAコマンドに応じて、フラッシュメモリ群4の有するプログラムコードの対応する処理ルーチンが個別に実行の対象となる。このようにすれば、将来に仕様の見直しの必要性が生じたり、オプションの追加などによって新たなATAコマンドのサ

ポートが必要になっても、ROMの焼き直しやRAMの容量のサイズアップをしなくても、容易に機能を追加することができる。

【0030】特にコントローラ8をワンチップで構成した場合、ROMの焼き直しやRAM容量のサイズアップはマスクの改定を伴う大がかりなものとなって容易には行えないため、本発明をするのに好適である。

【0031】もちろん本発明はATAカードのみに寄与するものではなく、その他RAMとフラッシュメモリとを備える情報処理装置であれば、同様に本発明を適用でき、その効果を得ることができる。

【0032】またフラッシュメモリとして、大容量のデータを蓄積できるように半導体ディスク装置を用いることもできる。

【0033】

【発明の効果】この発明のうち請求項1にかかる半導体記憶装置によれば、ファームウェアの修正や機能の追加を行う場合、不揮発性半導体記憶部の修正を行うことによって対応できるので、ROMの焼き直しを行う必要がない。しかもファームウェアの修正や機能の追加によって得られたプログラムのサイズが増大しても、RAMの第1の格納領域の容量を増大させる必要がないので拡張性に優れており、また少ないRAMの容量で多くの機能を実装できることで低価格を招来する。しかも、不揮発性半導体記憶部からRAMへとプログラムを転送する際に誤り訂正処理を施すので、訂正不可能な誤りが発生したプログラムのRAMへのロードを排除することができる。

【0034】この発明のうち請求項2にかかる半導体記憶装置によれば、複数のプログラムコードがロードされる前にその先頭アドレスを第2の格納領域に格納することにより、第1の格納領域に複数のプログラムコードの一つを格納する度に、不揮発性半導体記憶部にアクセスする必要がなく、迅速な動作が得られる。

【0035】この発明のうち請求項3にかかる半導体記憶装置によれば、仕様の見直し、オプションの追加などによって新たなコマンドをサポートする必要があるとしても、ROMの焼き直し、RAM容量のサイズアップを行うことなく、容易に機能を追加することができる。

【図面の簡単な説明】

【図1】 本発明の実施の形態にかかる半導体記憶装置の構成を示すブロック図である。

【図2】 本発明の実施の形態におけるRAMマップとフラッシュメモリマップとの関係を示した概念図である。

【図3】 本発明の実施の形態の動作を示すフローチャートである。

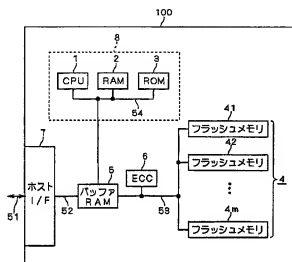
【図4】 従来の技術におけるRAMマップとフラッシュメモリマップとの関係を示した概念図である。

【符号の説明】

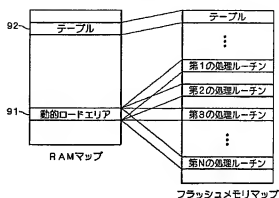
2 RAM、4 フラッシュメモリ群、41~4m フラッシュメモリ、5 バッファRAM、6 誤り訂正処理

回路、91 第1の格納領域、92 第2の格納領域。

【図1】



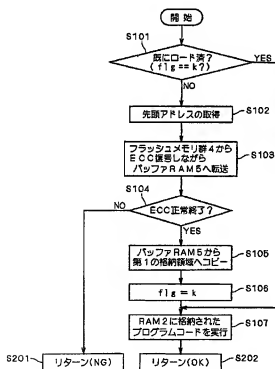
【図2】



91 : 第1の格納領域

92 : 第2の格納領域

【図3】



【図4】

